K 6

```
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEE              RRR         RRR  FFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEEEEEEEEEEE     RRRRRRRRRRRR     FFFFFFFFFFFF
EEE              RRR   RRR        FFF
EEE              RRR   RRR        FFF
EEE              RRR   RRR        FFF
EEE              RRR      RRR     FFF
EEE              RRR      RRR     FFF
EEE              RRR      RRR     FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
EEEEEEEEEEEEEEE  RRR         RRR  FFF
```

```
_$

Va
--
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
000
00
00
00
00
00
```

```
**FILE**ID**SBI                                   C  1




  SSSSSSSS  BBBBBBBB      IIIIII
  SSSSSSSS  BBBBBBBB      IIIIII
SS          BB      BB      II
SS          BB      BB      II
SS          BB      BB      II
SS          BB      BB      II
  SSSSSS    BBBBBBBB        II
  SSSSSS    BBBBBBBB        II
        SS  BB      BB      II
        SS  BB      BB      II
        SS  BB      BB      II            ....
        SS  BB      BB      II            ....
SSSSSSSS    BBBBBBBB      IIIIII          ....
SSSSSSSS    BBBBBBBB      IIIIII          ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
ER
Ps

PS
--

$E

Ph
--
In
Co
Pa
Sy
Pa
Sy
Ps
Cr
As

Th
79
Th
15
1

Ma
--
_$

0

Th

MA
```

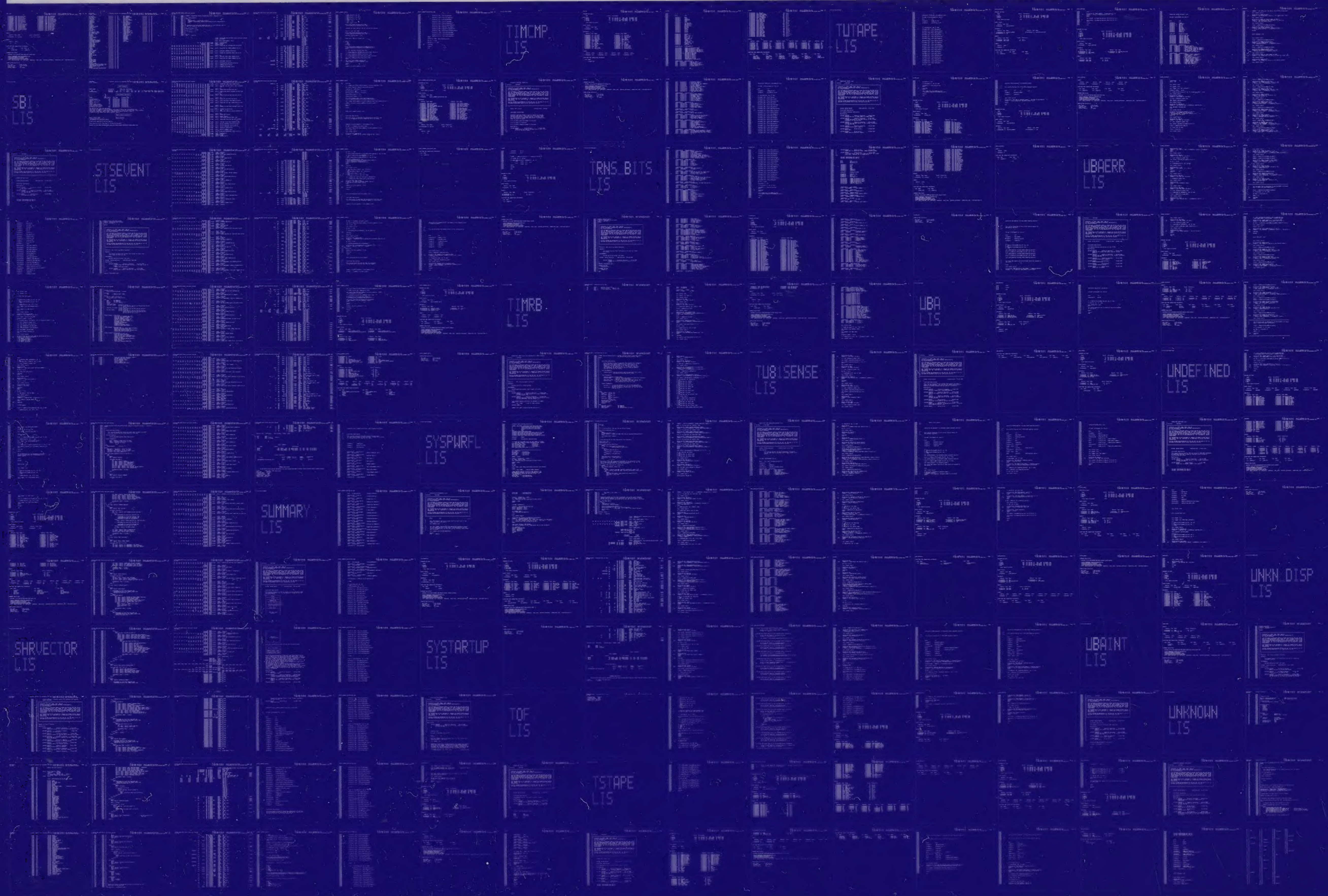
```
C
C Version:      'V04-000'
C
C****************************************************************************
C*                                                                          *
C*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
C*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
C*  ALL RIGHTS RESERVED.                                                    *
C*                                                                          *
C*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
C*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
C*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
C*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
C*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
C*  TRANSFERRED.                                                            *
C*                                                                          *
C*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
C*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
C*  CORPORATION.                                                            *
C*                                                                          *
C*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
C*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
C*                                                                          *
C*                                                                          *
C****************************************************************************
C

        SUBROUTINE SBI (LUN)

C
C       AUTHOR  BRIAN PORTER            CREATION DATE   27-AUG-1979
C
C++
C       Functional desciption:
C
C       Modified by:
C
C       V03-003 EAD0001         Elliott A. Drayton      18-Feb-1984
C               Add UVAX-1 support.
C
C       V03-002 SAR0096         Sharon A. Reynolds,     20-Jun-1983
C               Changed the carriage control in the 'format' statements
C               for use with ERF.
C
C       v03-001 BP0001          Brian Porter,           05-APR-1982
C               Corrected sbi alert bug.
C**
C--


        INCLUDE 'SRC$:MSGHDR.FOR /NOLIST'
        INCLUDE 'SRC$:SYECOM.FOR /NOLIST'
        BYTE            LUN

        INTEGER*4       SBI_FAULT

        INTEGER*4       SBI_COMP

        INTEGER*4       SBI_MAINT

        INTEGER*4       SBI_ERR

        INTEGER*4       SBI_TO

        INTEGER*4       SILO(0:15)

        INTEGER*4       SBI_REGA(0:15)

        INTEGER*4       ERROR_PC_780

        INTEGER*4       ERROR_PSL_780

        integer*4       error_pc_750

        integer*4       error_psl_750

        INTEGER*4       FIELD

        EQUIVALENCE     (SBI_FAULT,EMB(16))

        EQUIVALENCE     (SBI_COMP,EMB(20))

        EQUIVALENCE     (SBI_MAINT,EMB(24))

        EQUIVALENCE     (SBI_ERR,EMB(28))

        EQUIVALENCE     (SBI_TO,EMB(32))

        EQUIVALENCE     (SILO,EMB(36))

        EQUIVALENCE     (SBI_REGA,EMB(100))

        EQUIVALENCE     (ERROR_PC_780,EMB(164))

        EQUIVALENCE     (ERROR_PSL_780,EMB(168))

        equivalence     (error_pc_750,emb(16))

        equivalence     (error_psl_750,emb(20))

        integer*4       memory_registers_uv1(0:4)

        equivalence     (memory_registers_uv1(0),emb(16))

        PARAMETER       ASYNC_WRITE = 7

        integer*4       compress4

        logical*1       diagnostic_mode


        CALL FRCTOF (LUN)

        call header (lun)

c
c       11/780, 782, 785 support
c

        if (
        1 lib$extzv(24,8,emb$l_hd_sid) .eq. 255
        1 .or.
        1 lib$extzv(24,8,emb$l_hd_sid) .eq. 1
        1 ) then

        if (emb$w_hd_entry .eq. '07'x) then

        call logger (lun,'ASYNCHRONOUS WRITE')
        else

        call logger (lun,'SBI FAULT')
        endif

        call linchk (lun,2)

        write(lun,10) error_pc_780
10      format(/' ',t8,'ERROR PC',t24,z8.8)

        call vaxpsl (lun,error_psl_780)

        diagnostic_mode = .false.

        if (iand(sbi_maint,'f05ff900'x) .ne. 0) diagnostic_mode = .true.

        if (.not. diagnostic_mode) then

        CALL SBI_FAULTREG (LUN,SBI_FAULT)

        CALL SBI_COMPARATOR (LUN,SBI_COMP)

        CALL SBI_MAINTENANCE (LUN,SBI_MAINT)

        CALL SBI_ERROR (LUN,SBI_ERR)

        CALL SBI_TIMEOUT (LUN,SBI_TO)
        else

        call linchk (lun,6)

        write(lun,28) sbi_fault,sbi_comp,sbi_maint,sbi_err,sbi_to
28      format(' ',t8,'SBIFS',t24,z8.8,/,
        1 t8,'SBISC',t24,z8.8,/,
        1 t8,'SBIMT',t24,z8.8,/,
        1 t40,'DIAGNOSTIC MODE',/,
        1 t8,'SBIER',t24,z8.8,/,
        1 t8,'SBITA',t24,z8.8)
```

```
                endif

                IF (JIAND(SBI_COMP,'A0000000'X) .NE. 0
                1 .OR.
                2 JIAND(SBI_FAULT,'10000'X) .NE. 0) THEN

                if (.not. diagnostic_mode) then

                CALL LINCHK (LUN,3)

                WRITE(LUN,30)
        30      FORMAT(/' ','SBI SILO LOCKED, DETAILED SUMMARY',/)

                DO 50,I = 0,15

                CALL SBI_SILO (LUN,SILO(I))

        50      CONTINUE
                else

                CALL LINCHK (LUN,3)

                WRITE(LUN,52)
        52      FORMAT(/' ','SBI SILO LOCKED',/)

                do 54,i = 0,15

                call linchk (lun,1)

                write(lun,53) silo(i)
        53      format(' ',t24,z8.8)

        54      continue
                endif
                ENDIF

                DO 80,I = 0,15

                IF (SBI_REGA(I) .NE. 0) THEN

                CALL LINCHK (LUN,2)

                WRITE(LUN,55) I
        55      FORMAT(/' ','ADAPTER TR# ',I<compress4 (i)>,'.')

                CALL CLASSIFY (LUN,SBI_REGA(I))
                ENDIF

        80      CONTINUE

        c
        c       11/750 support
        c

                else if (lib$extzv(24,8,emb$l_hd_sid) .eq. 2) then

                if (emb$w_hd_entry .eq. '07'x) then

        call logger (lun,'WRITE BUS ERROR')
        endif

        write(lun,10) error_pc_750

        call vaxpsl (lun,error_psl_750)

c
c       UVAX-1 support
c

        else if (lib$extzv(24,8,emb$l_hd_sid) .eq. 7) then

        if (emb$w_hd_entry .eq. 7) then

        call logger (lun,'ASYNCHRONOUS WRITE')

        do 85,i = 1,16

        if (lib$extzv(15,1,memory_registers_uv1(i)) .eq. 1) then

        call memory_register_uv1 (lun,memory_registers_uv1)
        endif

85      continue

        endif

c
c       The IF-THEN-ELSE must be expanded at this point
c       to provide additional CPU "ASYNCHRONOUS WRITE
c       ERROR" support.
c

        endif

        RETURN



        ENTRY B_SBI (LUN)



        call header (lun)

        if (
        1 lib$extzv(24,8,emb$l_hd_sid) .eq. 255
        1 .or.
        1 lib$extzv(24,8,emb$l_hd_sid) .eq. 1
        1 ) then

        if (emb$w_hd_entry .eq. '07'x) then

        call logger (lun,'ASYNCHRONOUS WRITE')
        ELSE

        call logger (lun,'SBI FAULT')
        ENDIF

        else if (lib$extzv(24,8,emb$l_hd_sid) .eq. 2) then

        if (emb$w_hd_entry .eq. '07'x) then

        call logger (lun,'WRITE BUS ERROR')
        endif
        endif

        RETURN

        END
```

```
PROGRAM SECTIONS

    Name                                   Bytes   Attributes

  0 $CODE                                    961   PIC CON REL LCL   SHR   EXE   RD NOWRT LONG
  1 $PDATA                                   294   PIC CON REL LCL   SHR NOEXE   RD NOWRT LONG
  2 $LOCAL                                   324   PIC CON REL LCL NOSHR NOEXE   RD   WRT LONG
  3 EMB                                      512   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG
  4 SYECOM                                    44   PIC OVR REL GBL   SHR NOEXE   RD   WRT LONG

    Total Space Allocated                   2135

ENTRY POINTS

    Address   Type  Name              Address   Type  Name

  0-0000032F        B_SBI           0-00000000        SBI

VARIABLES

    Address   Type  Name                                    Address   Type  Name

  4-00000012  L*1   CP_11750                             4-00000011  L*1   CP_11780
  4-00000013  L*1   CP_117ZZ                             4-00000014  L*4   CRYPTK_FLAG
  4-0000000D  I*4   DEV_CHAR                             2-00000000  L*1   DIAGNOSTIC_MODE
  3-00000000  I*4   EMB$L_HD_SID                         3-00000004  I*2   EMB$W_HD_ENTRY
  3-0000000E  I*2   EMB$W_HD_ERRSEQ                      4-0000001E  L*1   END_VALUE
  4-0000001D  L*1   EOF_FLAG                             3-00000010  I*4   ERROR_PC_750
  3-000000A4  I*4   ERROR_PC_780                         3-00000014  I*4   ERROR_PSL_750
  3-000000A8  I*4   ERROR_PSL_780                        2-00000004  I*4   FIELD
  4-00000004  L*4   FORMS                                2-00000008  I*4   I
  4-0000000C  L*1   LINES                                4-00000027  I*4   LSTLUN
 AP-00000004@ L*1   LUN                                  4-0000001F  I*4   MAILBOX_CHANNEL
  4-0000002B  CHAR  OPTIONS                              4-00000008  L*4   PRINTER
  4-00000000  I*4   RECCNT                               4-00000023  I*4   RECORD_SIZE
  3-00000014  I*4   SBI_COMP                             3-0000001C  I*4   SBI_ERR
  3-00000010  I*4   SBI_FAULT                                 3-00000018  I*4   SBI_MAINT
  3-00000020  I*4   SBI_TO                                    4-00000019  L*1   VALID_CLASS
  4-0000001A  L*1   VALID_CPU                                 4-0000001B  L*1   VALID_ENTRY
  4-0000001C  L*1   VALID_TYPE                                4-00000018  L*1   VOLUME_OUTPUT


ARRAYS

    Address   Type  Name                                   Bytes  Dimensions

  3-00000000  L*1   EMB                                      512  (0:511)
  3-00000006  I*4   EMB$Q_HD_TIME                              8  (2)
  3-00000010  I*4   MEMORY_REGISTERS_UV1                      20  (0:4)
  3-00000064  I*4   SBI_REGA                                  64  (0:15)
  3-00000024  I*4   SILO                                      64  (0:15)


LABELS

    Address   Label       Address   Label       Address   Label       Address   Label       Address   Label       Address   Label

  1-00000049  10'      1-0000005F  28'      1-000000C1  30'             **      50      1-000000EA  52'      1-00000101  53'
          **  54       1-0000010A  55'              **  80              **      85


FUNCTIONS AND SUBROUTINES REFERENCED

  Type  Name                                Type  Name                                Type  Name

        CLASSIFY                            I*4   COMPRESS4                                 FRCTOF
        HEADER                              I*4   LIB$EXTZV                                 LINCHK
        LOGGER                                    MEMORY_REGISTER_UV1                       SBI_COMPARATOR
        SBI_ERROR                                 SBI_FAULTREG                              SBI_MAINTENANCE
        SBI_SILO                                  SBI_TIMEOUT                               VAXPSL


COMMAND QUALIFIERS

  FORTRAN /LIS=LIS$:SBI/OBJ=OBJ$:SBI MSRC$:SBI

  /CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
  /DEBUG=(NOSYMBOLS,TRACEBACK)
  /STANDARD=(NOSYNTAX,NOSOURCE_FORM)
  /SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)
  /F77  /NOG_FLOATING  /I4  /OPTIMIZE  /WARNINGS  /NOD_LINES  /NOCROSS_REFERENCE  /NOMACHINE_CODE  /CONTINUATIONS=19


COMPILATION STATISTICS

  Run Time:           3.86 seconds
  Elapsed Time:       11.88 seconds
  Page Faults:        179
  Dynamic Memory:     196 pages
```

0154 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



TIMCMP
LIS

TUTAPE
LIS

SBI
LIS

SYSEVENT
LIS

TRNS_BITS
LIS

UBAERR
LIS

TIMRB
LIS

UBA
LIS

TU81SENSE
LIS

UNDEFINED
LIS

SYSPWRFL
LIS

SUMMARY
LIS

UNKN_DISP
LIS

SHRVECTOR
LIS

SYSTARTUP
LIS

UBAINT
LIS

TOF
LIS

UNKNOWN
LIS

TSTAPE
LIS